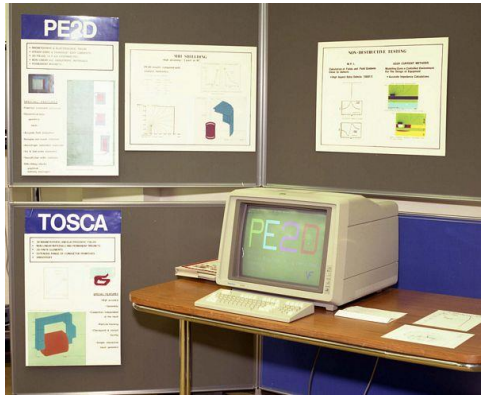


TOSCA / PE2D, RAL (UK), 1986

Demonstration, 1988

Opera 2D update and OperaPy

Round-table meeting for magnet designers

Kieran Geiger (TE-MSC-NCM), Ambre Visive (BE-EA-PR)

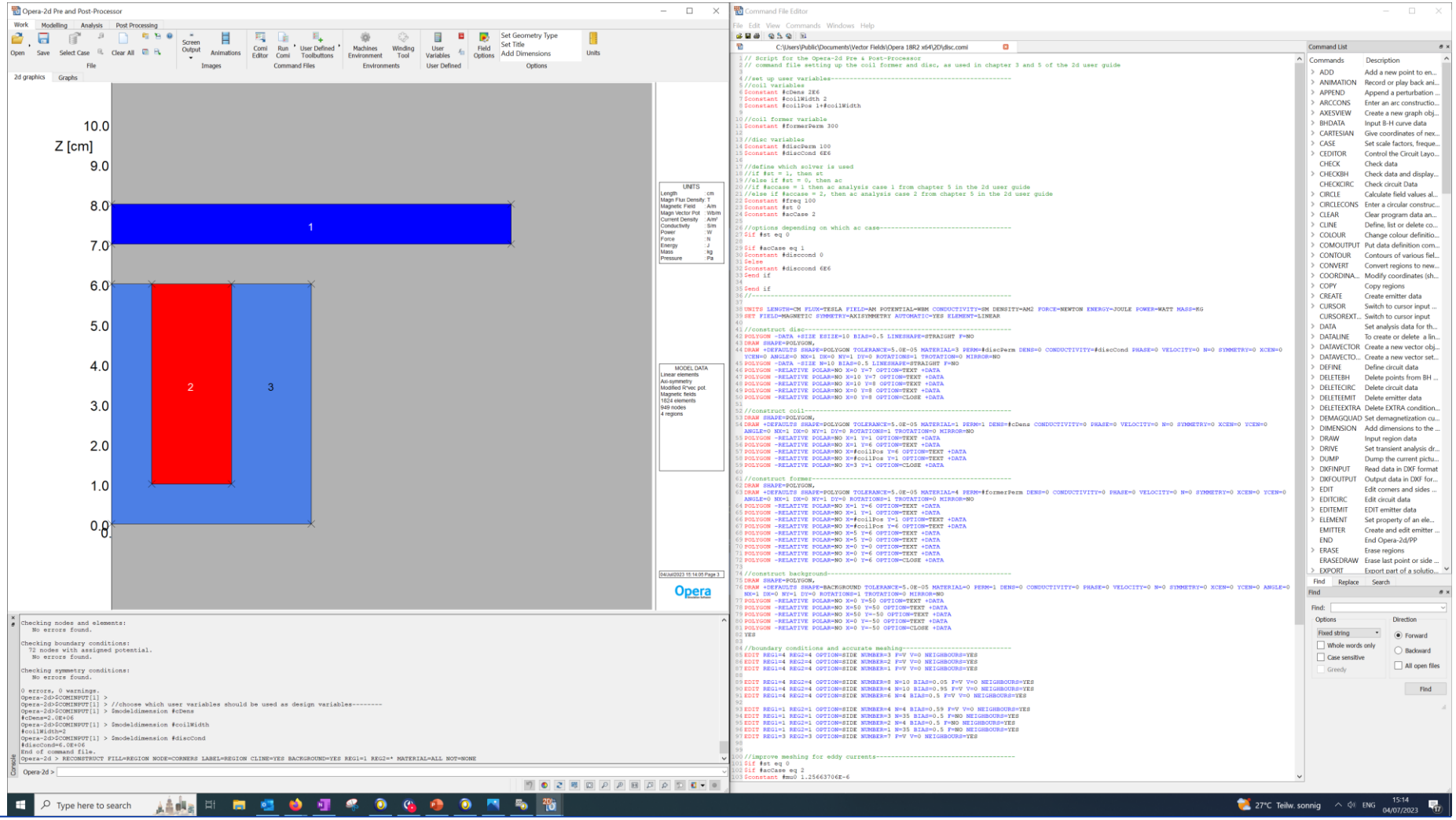
5th July 2023

Overview

1. **Demonstrations** (KG)
    1) Opera 2D 2018 and 2021+
    2) Python Integrated Design Environment (IDE)
    3) COMI script conversion
2. **Presentation:** “What could OOP and GIT do for you?” (AV)
3. **Strengths and Weaknesses** (KG)
4. **Group Discussion**
    - Magnet designer requirements
    - Towards an action plan (provisional)

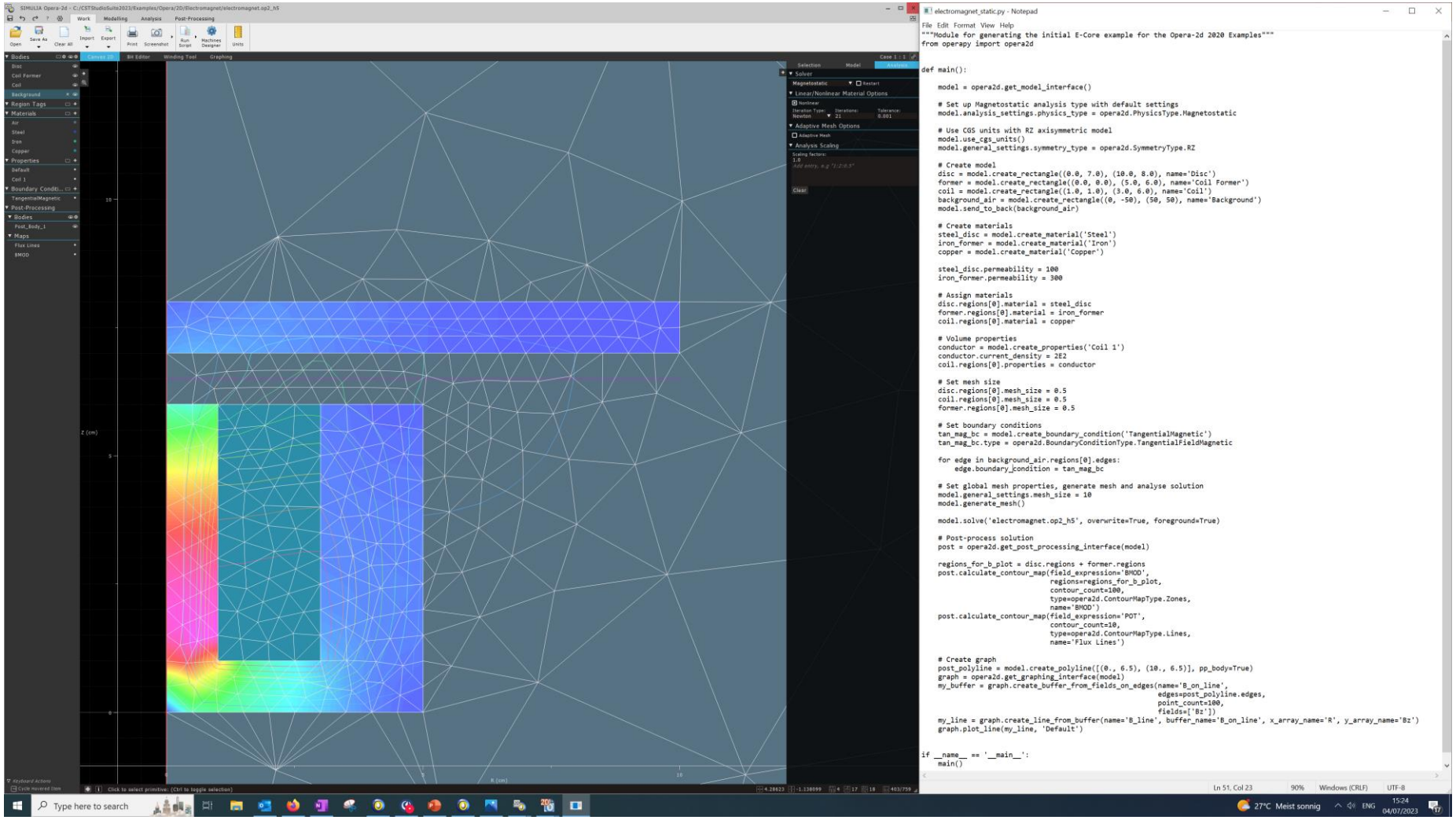
Demonstration: Opera 2D ≤2018

- Models can be generated using GUI or COMI scripting
- Graphical output has integrated COMI editor
- Convenient to create models by recording COMI commands from console



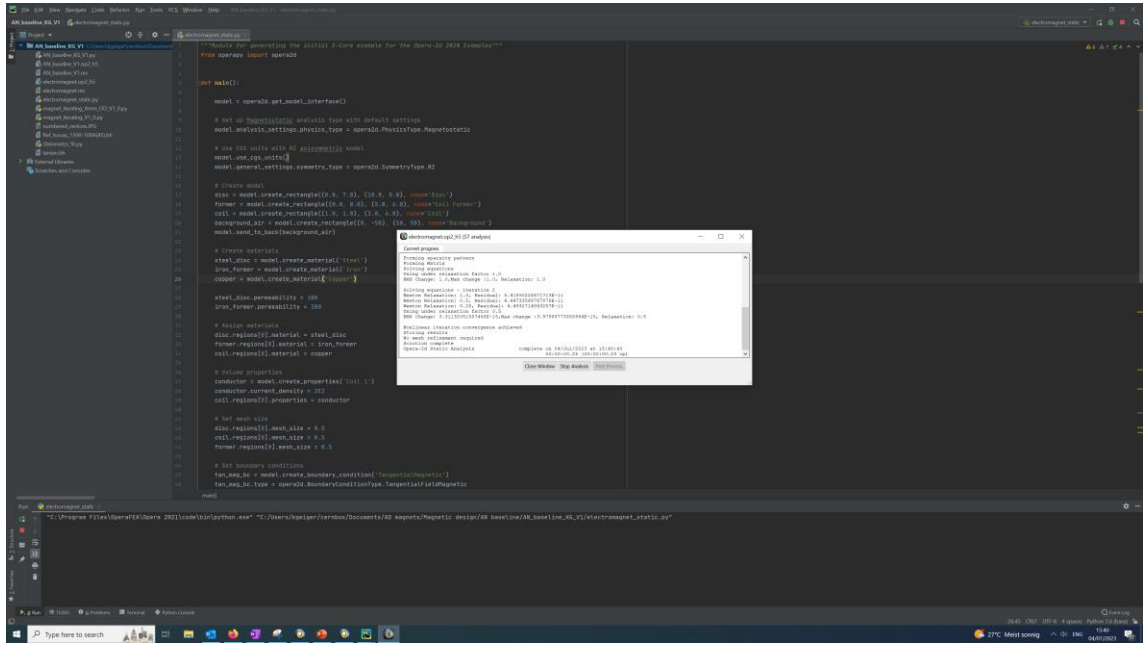
Demonstration: Opera 2D ≥2021

- Models can be generated using GUI, Python script, or legacy COMI script *
- There is a Python console but no editor **
- No longer practical to create models in GUI and then export script ***



Demonstration: Python IDE

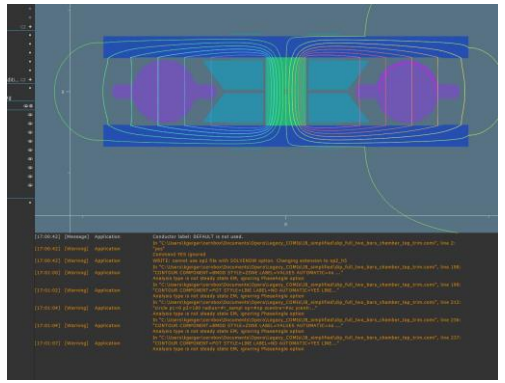
- **Integrated Design Environments**
  - Tool for code development. Includes Python interpreter, source-code editor, console, and debugger.
- **Opera supports use of 3rd party IDEs for developing code ***
  - No way to integrate IDE and graphical output
- **Programming paradigm** (Chat GPT)**:**
  - COMI (FORTRAN based): FORTRAN follows a procedural approach, focusing on writing step-by-step procedures or subroutines to perform tasks.
  - Python (with OOP): Python's OOP approach organizes code around objects and their interactions, making it easier to reuse and modify code.



Demonstration: COMI Script Conversion

Process

1. **Run COMI script using Opera 2D 2021+ COMI executor**
2. **Note that much of post-processing functionality does not work**
3. **Keep ‘compatible extract’ of COMI file**
4. **Recreate non-backwards-compatible COMI commands with OperaPy**
   - Taking advantage of Opera 2018 documentation and log files to better understand how COMI commands worked and what inputs/outputs they require/generate
5. **Create Python script which runs COMI extract and OperaPy commands together**



The CIRCLE Command

Summary

Calculate and graph fields on a circular arc.

Toolbutton

Command line parameters

Command	CIRCLE	
Parameter	Default	Function
RADIUS	*none*	Radius of arc.
P1	*none*	Azimuthal coordinate at start of arc.
P2	*none*	Azimuthal coordinate at end of arc.
XCENTRE	0	X coordinate at centre of arc.
YCENTRE	0	Y coordinate at centre of arc.
NP	100	Number of steps along the arc, i.e. NP+1 points.
COMPONENT	POT	Expression of field components.
TIME	0	For Steady-State AC solutions only: time at which basic field quantities are evaluated.
	number	Angle in degrees around ac cycle.
	AMPLITUDE	Amplitude.
	PHASE	Phase angle.
	TAVERAGE	Time-average.

Ambre Visive Presentation:
“What could OOP and GIT do for you?”

Opera 2D Weaknesses and Issues

1. **Opera's Python distribution is not robust**
   - Opera 2023 cannot be linked to 3rd party IDEs (reliant on 2021 version)
   - Not possible to link IDE and GUI
   - Export Python command (from GUI) is less useful than COMI equivalent
2. **Legacy COMI script executor**
   - Some COMI functions have been lost but this is not well documented
   - Does not raise useful errors or output log files
   - Very difficult to feed/fetch parameters to/from COMI scripts
3. **Opera technical support**
   - Documentation could be better
   - Support portal is good for narrow queries about EM simulation
     - E.g., methods for evaluating forces, formatting field plots, etc.
   - Slow responding to tickets about software issued. Unwillingness to acknowledge problems is frustrating

Opera 2D Strengths and Potential

- **Established tool at CERN – lots of experience and library of models**
  - Only useful if there is backwards COMI compatibility!
- **Compared to FEMM Opera has some additional features:**
  - Non steady state dynamic solver
  - Optimiser module
  - Hysteresis solver
- **Python interface**
  - Amber has given us a taste of its potential
  - Already popular with engineers and easy to learn

Group Discussion: Magnet Designer Requirements

- **Wants / needs from Opera 2D?**
  - Immediate needs
    - E.g., continued access to some Opera 2018 licenses for COMI debugging
  - Brainstorming ideas
    - E.g., repository of converted COMI scripts
- **Please start by quickly introducing yourself**
  - Previous experience with Opera 2D, COMI, and Python
  - What might you use Opera 2D for?

Group Discussion: Conclusions

- **From our discussion are there any clear priorities?**
  1. Immediate concerns
  2. Medium term (wish list)
- **Best way to work with Dassault/Opera?**
- **Going forwards:**
  - Is it useful to set up a quarterly meetings?
  - How could longer term projects be resourced?
- **Any other thoughts / business?**

home.cern